Protocole CMIA v1

1. Objet du protocole

CMIA est un protocole blockchain de type Proof of Work conçu pour maintenir un registre public, ordonné et vérifiable des transactions numériques. Le protocole a pour objectif de permettre le transfert de valeur, l'émission monétaire contrôlée, la validation distribuée des blocs et l'ouverture progressive du minage public. Son design privilégie la lisibilité du système, l'accessibilité technique et l'intégration directe avec l'infrastructure CryptoMonnaiesIA.

2. Identité réseau

Le protocole définit un réseau nommé CMIA Network, identifié par cmia-mainnet-v1, dont le symbole natif est CMIA. Chaque transaction et chaque mécanisme d'émission sont liés à cette identité réseau afin d'éviter l'ambiguïté entre environnements ou versions futures. Le protocole prévoit aussi une version de transaction TX_VERSION = 1, ce qui pose une base claire pour d'éventuelles évolutions compatibles ou ruptures de version ultérieures.

3. Modèle comptable

CMIA repose sur un modèle UTXO. La valeur n'est pas représentée comme un simple solde stocké dans un compte, mais comme un ensemble de sorties non dépensées rattachées à une clé publique. Une sortie peut être consommée une seule fois comme entrée d'une nouvelle transaction. L'état économique du réseau découle donc de l'ensemble des UTXO non marqués comme dépensés. Ce choix permet une validation explicite des flux, une traçabilité directe des sorties et une prévention structurelle de la double dépense. L'API expose d'ailleurs cet état via /utxo.

4. Transactions

Une transaction CMIA est identifiée par un hash dérivé d'un assemblage déterministe de champs comprenant l'émetteur, le destinataire, le montant, la signature et la date de création. Le protocole prévoit également des transactions strictes multi-input / multi-output, avec inclusion des entrées référencées, des sorties créées, des frais, de la version de transaction et du network_id. Dans le cas minier, une transaction spéciale de type coinbase est créée par le réseau avec sender_pub = "NETWORK" et attribue au mineur la récompense monétaire et les frais du bloc.

5. Signature et authentification

Le protocole utilise secp256k1 pour la signature et la vérification cryptographique. Ce choix s'inscrit dans la continuité des standards blockchain éprouvés et permet à un nœud de vérifier qu'une transaction a bien été autorisée par le détenteur légitime de la clé privée associée à la clé publique émettrice. La signature constitue ainsi le mécanisme fondamental d'authentification économique dans CMIA.

6. Structure d'un bloc

Un bloc CMIA contient au minimum les champs suivants : height, prev_hash, transactions, nonce, timestamp, difficulty et hash. Le hash du bloc est calculé à partir d'une sérialisation JSON ordonnée des données du bloc. Cette approche garantit qu'un même contenu produit un même hash, et qu'une modification même mineure du contenu invalide l'empreinte résultante. La chaîne est ainsi constituée d'une suite de blocs reliés cryptographiquement par référence au hash du bloc précédent.

7. Formation d'un bloc candidat

Lorsqu'un mineur déclenche le minage via POST /mine, le nœud rassemble les transactions admissibles, applique les limites de taille et de nombre de transactions, calcule les frais agrégés, puis construit un bloc candidat. Si une récompense peut encore être émise compte tenu de la supply restante, une transaction coinbase est insérée en première position dans la liste des transactions. Le bloc candidat reçoit alors une hauteur, une référence prev_hash, un horodatage, une difficulté et un nonce initial avant d'être soumis au calcul de preuve de travail.

8. Preuve de travail

CMIA utilise un mécanisme de Proof of Work. Le bloc candidat est miné par une routine qui recherche un nonce produisant un hash conforme à la difficulté en vigueur. La difficulté initiale est paramétrable, avec une valeur par défaut de 3, mais le protocole prévoit ensuite un ajustement dynamique. La preuve de travail sert à rendre coûteuse la production d'un bloc et à donner un ancrage objectif au consensus. Un bloc non conforme à la difficulté requise est rejeté.

9. Ajustement de difficulté

Le protocole fixe un temps cible de 30 secondes par bloc. La difficulté est recalculée tous les DIFFICULTY_ADJUST_EVERY_BLOCKS, soit tous les 5 blocs dans la configuration actuelle. Le recalcul compare le temps réellement observé sur la fenêtre de blocs à la durée attendue. Le ratio est borné entre 0,25 et 4,0 pour éviter les oscillations excessives. La nouvelle difficulté est ensuite obtenue par ajustement progressif à partir de la difficulté courante, tout en respectant une difficulté minimale. Ce mécanisme vise à stabiliser le rythme de production même en cas de variation de puissance de calcul.

10. Émission monétaire

La politique monétaire CMIA repose sur trois paramètres fondamentaux : INITIAL_REWARD = 25, HALVING_EVERY_BLOCKS = 5000 et MAX_SUPPLY = 10000000. La récompense est recalculée en fonction de la hauteur du bloc, selon un mécanisme de halving entier avec plancher à 1 unité. Pour chaque bloc miné, le nœud calcule la reward de base encore autorisée, la compare à la supply restante, puis définit reward_to_mint = min(base_reward, remaining_supply). Le montant final versé au mineur est coinbase_amount = reward_to_mint + total_fees. Ainsi, la création monétaire décroît dans le temps, tandis que les frais peuvent prendre progressivement plus de poids dans la rémunération du minage.

11. Validation des transactions

Lors du minage, les transactions intégrées au bloc sont marquées comme confirmées et associées au hash du bloc. Les UTXO consommées sont marquées comme dépensées, et de nouvelles UTXO sont créées pour chaque sortie valide des transactions confirmées. Si une UTXO censée être consommée ne peut pas l'être correctement, le nœud lève une erreur et le processus de validation échoue. Cette logique assure que chaque unité de valeur est consommée au plus une fois et que l'état UTXO reste cohérent avec l'historique de la chaîne.

12. Validation des blocs reçus

Le protocole permet aussi la réception de blocs externes via /block/receive. Un bloc reçu doit contenir les champs obligatoires, ne pas être déjà connu, et passer validate_block_full. Après validation, il est inséré dans la base, l'état réseau est recalculé, puis l'ensemble UTXO est reconstruit à partir de la chaîne si nécessaire. Cette étape garantit que l'acceptation d'un bloc externe ne repose pas sur la confiance, mais sur la validation complète de sa structure et de sa cohérence protocolaire.

13. État réseau

Le protocole maintient un état réseau persistant comprenant au minimum coins_minted, current_reward, max_supply, halving_every_blocks et la hauteur courante. Cet état est exposé via /state et permet de suivre l'émission réelle, la reward en vigueur et la progression du réseau. Il complète les autres endpoints de lecture comme /chain, /utxo et /health, qui rendent le protocole observable par les utilisateurs, les wallets, les explorers et les futurs mineurs publics.

14. Auto-miner et continuité du réseau

Le comportement de minage opérationnel actuel est défini par l'auto-miner. Celui-ci interroge la mempool à intervalles réguliers de 30 secondes. Si des transactions sont détectées, il déclenche immédiatement POST /mine. En l'absence de transactions, il attend jusqu'à EMPTY_BLOCK_INTERVAL = 600 secondes, puis lance un minage de bloc vide de

maintenance. Ce point est important : dans CMIA, la chaîne peut continuer à progresser même en faible trafic, ce qui maintient un rythme minimum de blocs, évite la stagnation du réseau et conserve une cadence d'émission et d'horodatage. Cette logique donne au protocole une dimension opérationnelle propre, distincte d'un système strictement passif.

15. Limites de capacité et règles de sûreté

Le protocole impose plusieurs plafonds : MAX_BLOCK_TX, MAX_BLOCK_BYTES, MAX_MEMPOOL_TX, MAX_TX_PER_SENDER_IN_MEMPOOL et MAX_TX_BYTES. Ces bornes visent à limiter les abus, à éviter la saturation mémoire, à contenir la taille des blocs et à réduire certaines formes de spam transactionnel. Elles font partie intégrante du protocole pratique, même si elles pourront évoluer selon les besoins du réseau.

16. Endpoints publics de référence

Dans son état actuel, le protocole est observable et actionnable à travers plusieurs endpoints publics : /health pour l'état du nœud, /state pour l'état économique, /chain pour la chaîne complète, /utxo pour l'ensemble des sorties, /mempool pour les transactions en attente, et /mine pour le déclenchement du minage côté nœud. Cette surface API fait partie de la matérialisation actuelle du protocole CMIA.

17. Règle de consensus pratique

Dans la forme actuelle du protocole, le consensus effectif est assuré par la validation locale du bloc, sa persistance en base, la mise à jour de l'état UTXO et la propagation aux pairs. Un bloc valide doit donc simultanément satisfaire les conditions structurelles, cryptographiques, monétaires et comptables du protocole. La chaîne retenue n'est pas une simple suite de données mais une suite de transitions d'état validées et compatibles avec l'économie du réseau.

18. Formulation courte du protocole

CMIA peut se résumer ainsi : un utilisateur crée une transaction signée ; la transaction entre en mempool ; un mineur déclenche la construction d'un bloc ; le nœud sélectionne les transactions admissibles, ajoute une coinbase, calcule une preuve de travail, écrit le bloc, marque les entrées consommées, crée les nouvelles sorties, met à jour l'état monétaire et diffuse le résultat au réseau. En l'absence de trafic, un auto-miner peut forcer périodiquement un bloc de maintenance afin de préserver la continuité opérationnelle de la chaîne.

19. Conclusion protocolaire

Le protocole CMIA v1 n'est pas seulement une idée théorique : il correspond déjà à une implémentation opérationnelle. Sa spécificité ne tient pas seulement à ses paramètres monétaires ou à son API, mais au fait qu'il combine un modèle UTXO, une émission

décroissante, un minage HTTP piloté par nœud et une logique d’auto-maintenance du réseau par blocs vides. Cela te donne une base solide pour écrire un whitepaper “style Bitcoin”, mais avec une identité CMIA propre.